

# **Sistemas embebidos 2.**

## **“Laboratorio 2:** *USB HID device***.”**

# **Profesor: Rodrigo Aldana López.**

# **Alumnos:**

# Gustavo Araiza Obeso ie699209.

# Celis Tapia Daniel ie689705.

# Fecha:

# 14/05/2018

## INTRODUCCION:

Como parte de la materia de Sistemas Basados en Microprocesadores II que imparte el profesor Rodrigo Aldana López, se llevó a cabo del desarrollo del segundo laboratorio de la materia, que consistió en la implementación de un sistema HID USB que funciona como mouse y teclado, realizando una serie de tareas. Este laboratorio fue realizado por los alumnos Gustavo Araiza Obeso y Daniel Celis Tapia.

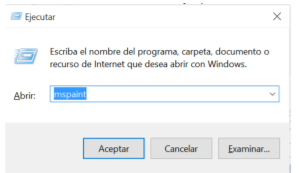
En el siguiente reporte se analizarán todos los aspectos que fueron necesarios para la elaboración de este laboratorio, empezando por el planteamiento del problema, en otras palabras, las especificaciones de esta.

Posteriormente se encuentra la sección de desarrollo, donde se explicará de manera detallada cueles fueron los pasos a seguir para el desarrollo de este mismo.

# Descripción funcional

Se desarrolla un sistema de basado en la FRDM-K64F que funciona como USB HID device. La tarjeta se conecta a la una PC con Windows a través del puerto K64F USB. Al conectarse a la PC, lo que el usuario observa es la siguiente secuencia de eventos:

• Se abre una ventana de “ejecutar”, y se escribe en ella “ms Paint” como se muestra en la siguiente figura:



• Se abre MS Paint y en él se dibuja una figura cerrada (un polígono, por ejemplo).

• Se abre un bloc de notas mediante el cuadro de dialogo “Ejecutar” y este se coloca en la mitad izquierda de la pantalla.

• Se abre otro bloc de notas y se coloca en la mitad derecha de la pantalla.

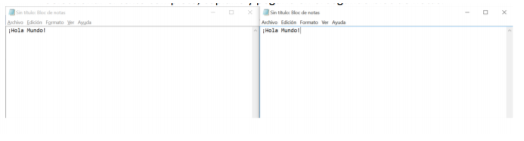
• El cursor se mueve para seleccionar el bloc de notas izquierdo.

• En el bloc de notas izquierdo se escribe el mensaje “¡Hola mundo!”.

• El mensaje se selecciona completo, y se copia.

• El cursor se mueve para seleccionar el bloc de notas derecho.

• El mensaje “¡Hola mundo!” se pega en el bloc de notas derecho. El resultado final se ve como en la siguiente figura:



## Desarrollo:

Para el desarrollo practico de este laboratorio, se inició con la sugerencia descrita en el archivo dado por el profesor, lo cual consistía en elaborar el report descriptor correspondiente para el funcionamiento adecuado del mouse y teclado en un mismo archivo, esto con la ayuda de una herramienta de igual manera proporcionada por el profesor. Continuando así a modificar la funcion de USB\_DeviceHidMouseAction para controlar la funcion de estos dos.

Para poder realizar esta acción, era necesario modificar el buffer para poder manejar el teclado y el mouse a través de este, esto como tal no fue de nuestro agrado, ya que ambas acciones se manejaban en la misma tarea, teniendo que esperar a que finalice una acción para comenzar otra.

Por lo que se optó por implementar de manera independiente estas acciones, realizando una tarea para el mouse y otra para el keyboard, así tendrán sus respectivos buffers y podremos manejarlas de forma paralela y al mismo tiempo.

Como ya sabíamos cómo era el funcionamiento del mouse (ejemplo visto en clase), se inicio a probar con el funcionamineto del teclado; una vez entendido, lo primero fue construir una máquina de estados donde cada estado correspondería a una acción a realizar por el teclado, lo primero fue hacer que se abriera la ventana “ejecutar” para poder abrir el Paint, esto fue relativamente sencillo ya que solo era mandar por el buffer las dos teclas correspondientes.

Cabe destacar que Windows solo puede soportar 6 teclas presionadas por lo que no se podía mandar mas de 6 teclas a través del buffer, por lo tanto, fue necesario ajustar los tiempo al momento de escribir en la ventana de ejecutar y algo muy importante a considerar, siempre después de usar el buffer, era necesario limpiarlo (asignarlo a cero) de lo contrario es como si se quedara presionada la tecla y al desconectar la kinetis la tecla quedaba presionada en la computadora y solo quedaba reiniciar.

Teniendo esta parte realizada, era necesario continuar con la parte del Mouse, que, mediante banderas, se le indicaría a en que momento entrar en acción, como primera acción a realizar, fue un centrado del cursor, no importa en qué posición se encuentre el cursor, este siempre será centrado y esta acción se ejecuta al mismo tiempo que se abre la herramienta Paint, para cunado se abra el cursor ya esté listo para presionar y comenzar a dibujar la figura.

El procedimiento anterior se repite, pero cambiando en la máquina de estados según sea la tarea a realizar ya que cada tarea requiere un tiempo y procedimiento especifico a realizar.

## Conclusiones:

***Gustavo:*** Esta práctica me pareció muy interesante, ya que no sabia todo lo que pasaba al momento de conectar un dispositivo al PC, todo lo que pasa en un par de segundos para poder utilizar disco dispositivo, me gustó mucho el tener que crear nuestros report descriptor para el mouse y teclado para poder utilizarlos, y ni se diga el tener que desarrollar de forma correcta las tareas realizadas sin causar un desastre en la PC. Algo que también me pareció interesante fue realice una pequeña subrutina en la tarea del mouse para al iniciar el sistema, el cursor se centre para poder tener un punto de origen, a lo que platicando con mis compañeros a ninguno se le había ocurrido esa acción. Para concluir el laboratorio, fue una excelente actividad el poder implementar en la tarjeta una forma de controlar a través del protocolo USB un dispositivo y asignarle tareas a realizar una vez conectado a la PC.

***Daniel***: A través del uso de dos descriptores y, por ende, de dos reportes, se logró crear la secuencia de comandos que permitieron ejecutar los requisitos del laboratorio exitosamente. Como puede verse, realmente dependen de una serie de condicionales para enviar las cadenas esperadas en el momento en que se necesitan. Además, se cuenta con la posibilidad de centrar el mouse. Estas ideas son crédito de mi compañero, y a través de ellas se puede ver la flexibilidad de mandar información a través del protocolo USB y cómo instanciar la tarjeta de desarrollo como una serie de dispositivos a los que responde exitosamente el sistema; ya que los reportes esperados mantienen la estructura genérica de un par de dispositivos HID.

# Repository link:

Este Proyecto puede ser encontrado en el siguiente link:

<https://github.com/GustavoAraizaO/Laboratorio2/tree/master/Laboratorio2_Mouse_keryboard>